

# Table des matières



- ▶ Chapitre 1: Introduction aux bases de données
- ▶ Chapitre 2: Le modèle Entité–Association
  1. Méthodologie de modélisation
  2. La notion d'Entité et de Classe d'Entité
  3. Les relations: l'Association
  4. Les Attributs
  5. Les relations: la Spécialisation–Généralisation
  6. Les contraintes d'intégrités
- ▶ Chapitre 3: Le Schéma Relationnel
- ▶ (Chapitre 4: SQL)

# Chapitre 2:

# Le modèle Entité–Association



1. Méthodologie de modélisation

# Créer une base de données "en live"

- ▶ On **ne** crée **pas** une base de données d'une entreprise directement dans le workbench d'un SGBD!
  - Pas de communication, ou difficilement
  - Travail solitaire
  - Pas de modèle conceptuel ni logique
  - Pas de travail itératif de modélisation
  - Pas de documentation de l'implémentation (le plus souvent)
  - Source d'erreurs importantes (Rappel: en informatique, plus une erreur est découverte tôt avant l'implémentation, moins elle a de conséquences financières et temporelles)
  - ...
- ▶ Nécessité d'une approche disciplinée, structurée et basée sur des techniques et modèles rigoureux

# Quelques chiffres

- ▶ Une base de données de taille moyenne:
  - comporte plusieurs centaines de tables
  - comporte plusieurs milliers de colonnes
- ▶ Une base de données de grande taille:
  - comporte plusieurs milliers de tables
  - comporte plusieurs dizaines de milliers de colonnes
- ▶ SAP utilise une BD de:
  - de près de 30.000 tables
  - de près de 200.000 colonnes

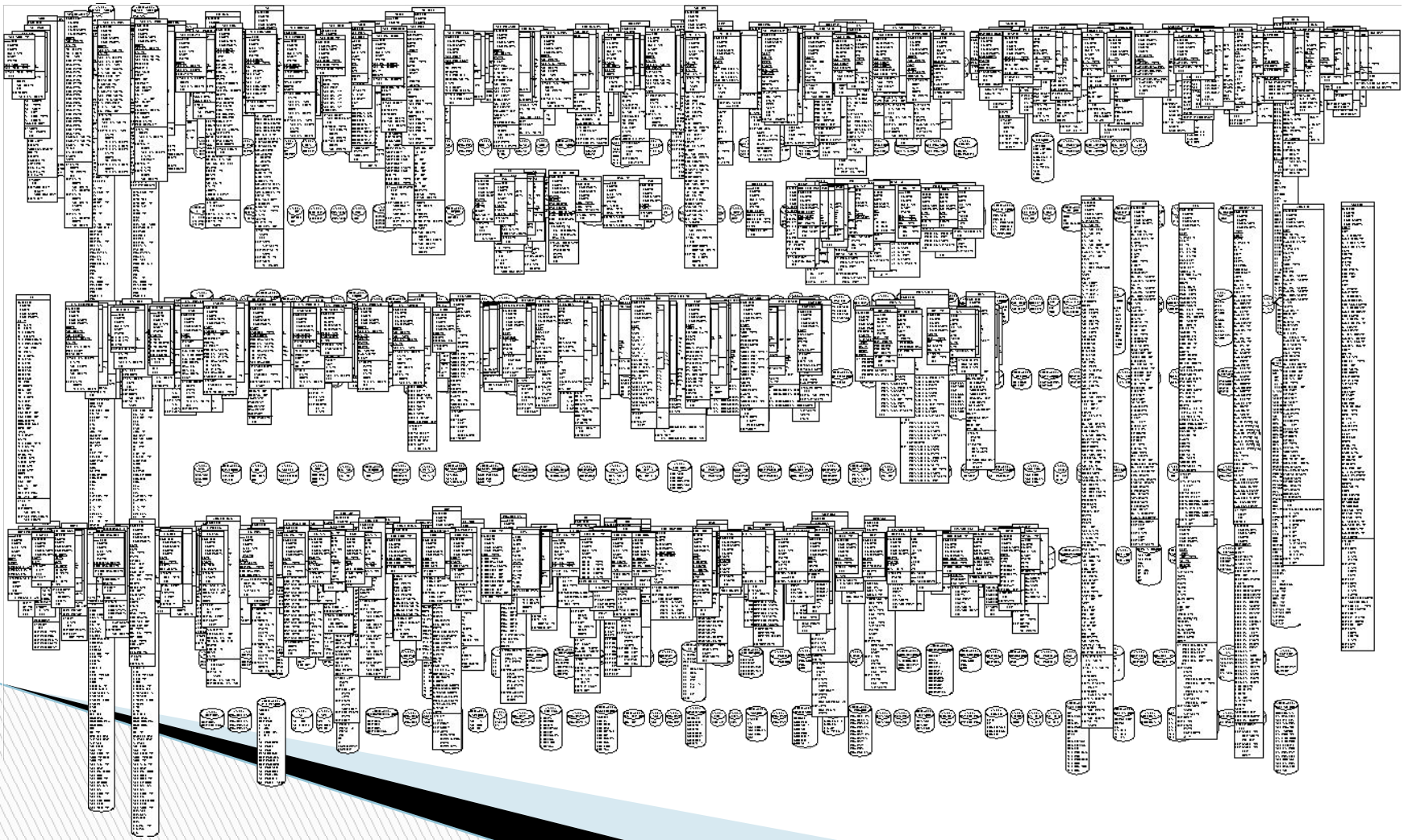
Sans modèles de différents niveaux, impossible de la comprendre!



# Une exemple réel (1999)

Entreprise européenne de vente par correspondance

- ▶ 382 espaces de stockages et 853 tables



# Le problème de modélisation

*Comment traduire les besoins exprimés par les utilisateurs en structures de données ?*

- ▶ Les utilisateurs = tous les agents (personnes, organismes, programmes et applications clients) impliqués dans la collecte, la production, la consultation, l'utilisation, la transmission, le traitement et la gestion des données
- ▶ Les besoins (ou exigences): ensemble des propriétés, caractéristiques et conditions que la base de données doit respecter pour satisfaire ses utilisateurs
  - besoins *fonctionnels* : contribuent à la fonction (au métier) des utilisateurs
  - besoins *non fonctionnels* : contribuent à la qualité des services rendus aux utilisateurs
- ▶ Les structures de données = le(s) schéma(s) de la base de données

# Modéliser en 2 (?) étapes

*Un ouvrage est une oeuvre littéraire publiée. Il est caractérisé par son numéro identifiant, son titre, son éditeur, sa date de première parution, ses mots-clés (10 au maximum), une brève note de présentation (ces notes sont en cours de constitution), le nom et le prénom de ses auteurs. A un ouvrage correspondent un certain nombre d'exemplaires, qui en sont la matérialisation physique. ...*

?

Conception de  
base de données

```
create database BIB
create dbspace BIB_DATA;
create table OUVRAGE (
    NUMERO char(18) not null,
    TITRE varchar(60) not null,
    EDITEUR char(32) not null,
    DATE_1RE_PARUTION date not null,
    PRESENTATION varchar(255),
    primary key (NUMERO)) in BIB_DATA;
. . .
alter table EXEMPLAIRE add constraint FKDE
    foreign key (NUMERO) references OUVRAGE;
. . .
create unique index IDOUVRAGE
    on OUVRAGE (NUMERO);
. . .
```

besoins des  
utilisateurs

?

Conception de  
base de données

code SQL  
DDL



# Méthodologie de développement de BD

Un ouvrage est une oeuvre littéraire publiée. Il est caractérisé par son numéro identifiant, son titre, son éditeur, sa date de première parution, ses mots-clés (10 au maximum), une brève note de présentation (ces notes sont en cours de constitution), le nom et le prénom de ses auteurs. A un ouvrage correspondent un certain nombre d'exemplaires, qui en sont la matérialisation physique. ...

Contenu de base de données

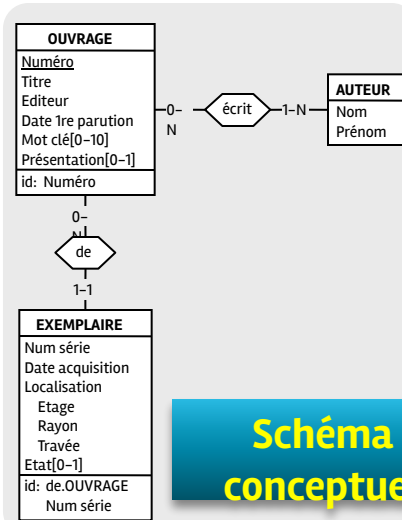


Schéma conceptuel

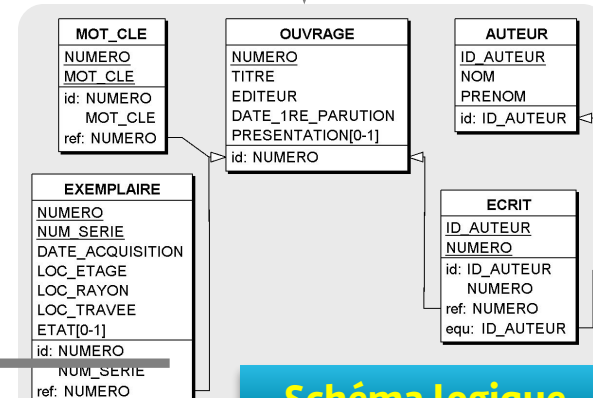


Schéma logique (relationnel)

```
create database BIB
create dbspac BIB DATA;
create table OUVRAGE (
  NUMERO char(18) not null,
  TITRE varchar(60) not null,
  EDITEUR char(32) not null,
  DATE_1RE_PARUTION date not null,
  PRESENTATION varchar(255),
  primary key (NUMERO)) in
BIB_DATA;
...
alter table EXEMPLAIRE add constraint
FKDE
foreign key (NUMERO)references
OUVRAGE;
...
create unique index IDOUVRAGE
on OUVRAGE (NUMERO);
...
```

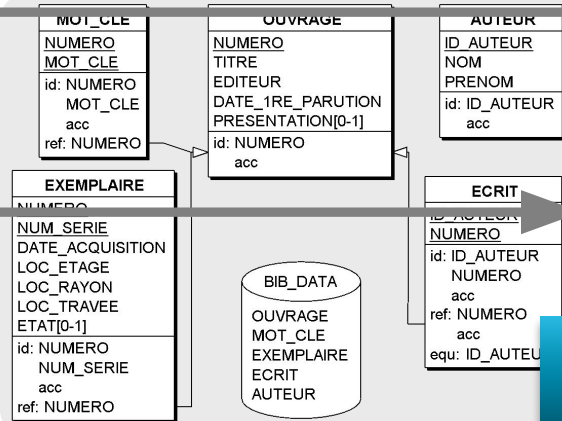


Schéma physique (Oracle 11)



# Modèle VS. Schéma

- ▶ Un **modèle**: un système formel de représentation de certains aspects de tous les domaines d'applications d'un certain type
  - Donne les constructeurs et la grammaire à suivre
  - En BD: représenter des objets, leurs attributs, leurs liens,...
- ▶ Un **schéma**: représentation des concepts statiques (et abstraits) d'un domaine d'application *particulier*
  - Un schéma peut être positionné avec différentes vues sur le domaine d'application
  - Un schéma peut être positionné à différents niveaux (ex: conceptuel, logique ou physique en BD)
- ▶ La population sont les différents instances d'un schéma particulier, à un moment particulier

# Chapitre 2:

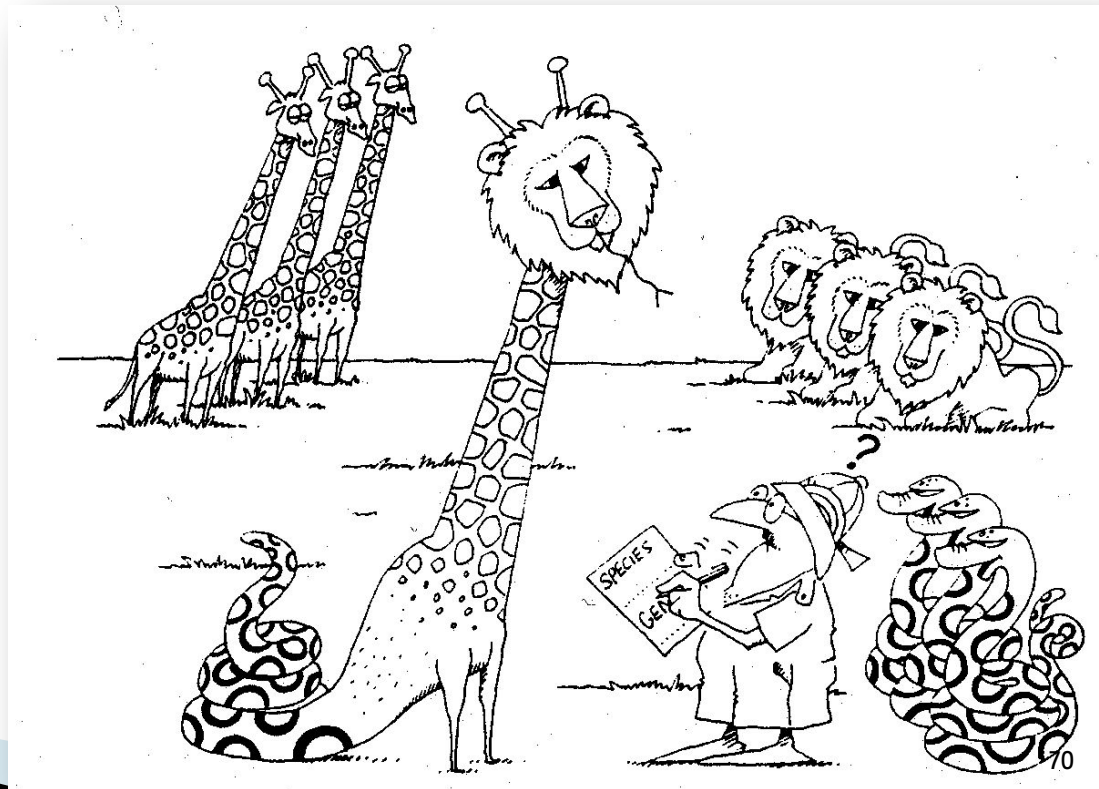
## Le modèle Entité–Association



2. La notion d'Entité et de Classe d'Entité

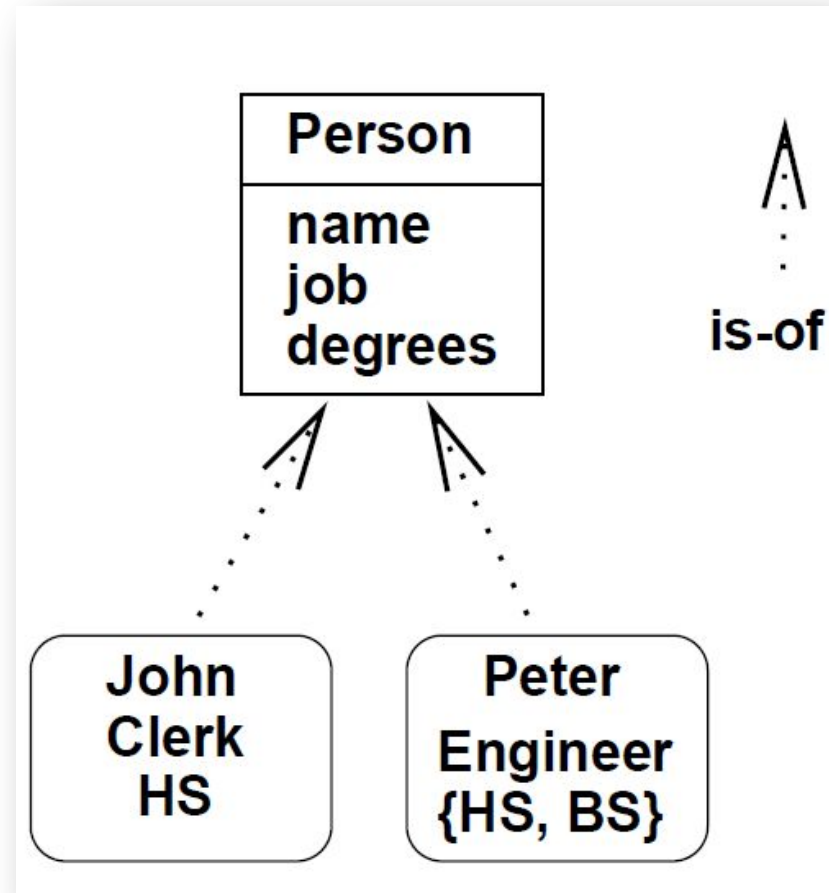
# L'Entité

- ▶ Une Entité correspond à un objet du monde réel
  - Entité physique: des personnes, des machines, des produits,...
  - Entité conceptuelle/abstraite: une société, un travail, une commande,...



# La Classe d'Entités

- ▶ Une Classe d'Entité
  - Concept générique dans lequel on classe/instancie plusieurs Entités ayant des caractéristiques similaires
  - Une Classe d'Entité définit un template commun pour un ensemble d'Entité



# Exercices



- Quelques exercices pour exploiter les concepts d'Entité et de Classe d'Entités



# Exercice 1

- ▶ Voici trois entités :
  - *Jules Dumoulin, 25 ans, habitant rue d'Erquelines n° 18 à 5983 Mont-sur-Pieds, marié, 1 enfant*
  - *Lisa Lambert, 78 ans, habitant rue d'Emailas n°134 à 1930 Souson, veuve, 4 enfants*
  - *Henri, cheveux bruns, yeux bleus, 187cm, 56 kilogrammes, vendeur de meubles*
- ▶ Identifiez l'entité ne faisant pas partie de la même classe que les autres  
Expliquez pourquoi

# Exercice 2

- ▶ Voici deux entités :
  - *Jules Dumoulin, 25 ans, habitant rue d'Erquelines n° 18 à 5983 Mont-sur-Pieds, marié, 1 enfant*
  - *Lisa Lambert, 78 ans, habitant rue d'Emailas n°134 à 1930 Souson, veuve, 4 enfants*
- ▶ Nommez la classe pouvant regrouper ces deux entités et donnez des caractéristiques partagées



# Exercice 3

- ▶ Voici la description d'une Classe d'Entités:  
Film(Titre, AnneeProd, ActeurPrincipal, Langue, Réalisateur, NbrePrix)
- ▶ Voici l'extrait d'un article; donnez l'entité correspondant à la classe d'entité ci-dessus:  
*Le film mythique « Dear Friends » produit par Universal dans le Kansas, Etats-Unis, en 1987 et sorti dans les salles l'année suivante (1988) vient encore de remporter un prix récemment. Ce prix, nouveau triomphe pour l'équipe de production, a été remis à la veuve du réalisateur, Thomy Lee. L'acteur José Despero était évidemment présent. Son rôle central dans le film a encore été souligné. Sa partenaire, Monique Poncin, était également présente, bien que souffrante. Ce film va probablement être adapté pour être joué au théâtre selon le producteur de la troupe théâtrale « Moments Heureux ».*

# Chapitre 2:

# Le modèle Entité–Association



## 3. Les Attributs

# Les attributs: définition

- ▶ Un **Attribut** d'une Classe est une caractéristique partagée par (toutes?) ses Entités
- ▶ Chaque Attribut a un domaine de valeurs (type) précis
- ▶ Chaque Entité a une valeur pour les (chaque?) attributs de sa Classe
- ▶ **Attribut atomique**

Personne
Nom
Job
Diplome

# La cardinalité d'un Attribut

- ▶ Une **cardinalité**: concept mathématique définissant le nombre d'élément (ou de relations) dans/entre ensembles finis
- ▶ Soit la cardinalité  $[x,y]$ 
  - $x$ : borne inférieure;  $y$ : borne supérieure
  - Donne le nombre min  $x$  et max  $y$  de valeur que peut prendre une Entité pour cet attribut
  - $x \leq y$
- ▶ **Optionnel VS. Obligatoire**
  - Optionnel:  $x = 0$
  - Obligatoire:  $x \geq 1$
- ▶ **Multiplicité** d'une cardinalité
  - Unique (une valeur):  $y = 1$
  - Multiple (plusieurs valeurs):  $y > 1$
- ▶ Par défaut:  $[1,1]$

Personne
Nom
Job
Diplome[0-N]
Tel[1-3]

Personne
Nom
Job
Diplome[0-N]

# La valeur NULL

- ▶ Si une cardinalité est optionnelle, alors on peut ne pas avoir de valeur pour cet attribut lors de l'enregistrement/modification d'une entité particulière
- ▶ *NULL* (le "rien informatique")
  - **Pas applicable:** ne s'applique pas à l'entité
  - **Inconnu actuellement, mais la valeur existe:** on sait que l'entité possède une valeur pour l'attribut sans encore la connaître
  - **Inconnu actuellement, la valeur existe peut-être:** on ne connaît pas la valeur et on ne sait pas si elle existe
- ▶ Concept central à toujours garder dans un coin de la tête lorsque l'on réalise des requêtes SQL DRL  
(et principalement des requêtes agrégatives, par ex. calculer une moyenne ou compter un nombre d'enregistrement)

# Attribut dérivé

- ▶ Si deux attributs sont en relation, ils sont appelés **attributs dérivés**
  - Ex: Age peut-être dérivé de DateNaissance
  - **Redondance!**
- ▶ Garder l'attribut le plus pérenne dans le temps

Personne
Nom
Job
Diplome[0-N]
DateNaissance
Age

# Attribut composite

- ▶ Un **Attribut composite** est un attribut qui est décomposé en d'autres attributs
- ▶ Un attribut composite est donc un ensemble de valeurs mise en relation
- ▶ Un attribut atomique est donc un attribut indivisible
- ▶ C'est un groupe de valeurs qui n'est pas suffisant que pour être une entité en tant que tel

Personne
<u>NumNational</u>
Nom
Job
Diplome[0-N]
DateNaissance
Adresse
Rue
Numero
CodePostal
Ville
id: NumNational



# Attribut identifiant

- ▶ Un **attribut identifiant** est un attribut (simple ou complexe) dont la valeur peut à elle seule identifier chaque entité de la Classe par rapport aux autres
  - **Unicité**: pas deux fois la même valeur pour deux entités dans la classe
  - **Valeur obligatoire** : pas de valeur NULL  
=>cardinalité: [1,1]
- ▶ Une entité peut avoir plusieurs identifiants, un seul est choisi comme attribut identifiant
- ▶ Un identifiant peut être composé de plusieurs attributs simples ou composites

Personne
<u>NumNational</u>
Nom
Job
Diplome[0-N]
DateNaissance
id: NumNational

Personne
<u>Nom</u>
Job
Diplome[0-N]
DateNaissance
<u>Adresse</u>
Rue
Numero
CodePostal
Ville
id: Nom Adresse

# Autres notations pour les différents types d'Attributs

